

⑫ 公開特許公報(A)

平1-147722

⑤ Int. Cl.⁴
G 06 F 9/38識別記号
310庁内整理番号
X-7361-5B

④ 公開 平成1年(1989)6月9日

審査請求 未請求 発明の数 1 (全5頁)

⑬ 発明の名称 情報処理装置のパイプライン処理方式

⑭ 特 願 昭62-308111

⑮ 出 願 昭62(1987)12月4日

⑯ 発 明 者 境 則 彰 東京都港区芝5丁目33番1号 日本電気株式会社内
 ⑰ 出 願 人 日本電気株式会社 東京都港区芝5丁目33番1号
 ⑱ 代 理 人 弁理士 内 原 晋

明 細 書

1. 発明の名称

情報処理装置のパイプライン処理方式

2. 特許請求の範囲

先行制御をおこなう情報処理装置において、演算実行のパイプライン処理をおこなわず、短いパイプライン動作をおこなう動作モードと、演算実行のパイプライン処理をおこない、長いパイプライン動作をおこなう動作モードとの動作モード遷移をおこなう情報処理装置のパイプライン処理方式。

3. 発明の詳細な説明

(産業上の利用分野)

本発明はパイプライン処理をおこなう情報処理装置に関し、特にパイプライン処理方式に関する。

(従来の技術)

従来、この種の情報処理装置においては、浮動小数点数命令、10進数命令の演算は演算が複雑なため、演算の実行に複数サイクルかけて演算をおこなっていた。また、スーパーコンピュータや

大型計算機のなかには演算装置もパイプライン化されているものもある。

(発明が解決しようとする問題点)

上述した従来の情報処理装置では、浮動小数点数命令、10進数命令等は演算の実行に複数のサイクルが必要なため、そのような命令を実行するとパイプラインに乱れが生じ命令実行のスループットが低下するという欠点があり、またスーパーコンピュータ等のように演算装置をパイプライン化すると、連続した演算実行が可能になるため、パイプラインに乱れは生じずスループットも低下しないが、反面演算処理サイクルがパイプライン化されただけパイプラインが長くなるため、分岐予測失敗の判定が遅れたり、アドレス修飾用レジスタの確定待ちが長くなるという欠点がある。

(問題点を解決するための手段)

本発明の情報処理装置のパイプライン処理方式は、演算実行のパイプライン処理をおこなわず、短いパイプライン動作をおこなう動作モードと演算実行のパイプライン処理をおこない、長いパイ

パイプライン動作をおこなう動作モードとの動作モード遷移をおこなう。

〔作用〕

浮動小数点命令の頻度が低く、分岐命令の頻度の高いオペレーティングシステム等のシステムプログラム動作時は、非演算パイプライン動作をおこなうことで出現頻度の高い分岐命令の処理時間を減らし、逆に浮動小数点命令の頻度が高く、分岐命令の頻度が低い科学技術計算プログラム等の動作時には演算パイプライン動作をおこなうことで、出現頻度の高い浮動小数点命令の処理時間を減らし、全体的なスループットを高めることができる。

〔実施例〕

次に、本発明の実施例について図面を参照して説明する。

第1図は本発明のパイプライン処理方式が適用された情報処理装置の一実施例のブロック図である。

本実施例は、命令制御装置1、記憶制御装置2、

ランドアドレスが生成され、記憶制御装置2に送られ、アドレスレジスタ(VAR)21に保持される。このオペランドアドレスはアドレス変換機構(TLB)22によりアドレス変換がおこなわれ、キャッシュのアドレスアレイ(AA)22より読出したディレクトリ情報とともにアドレスレジスタ(PAR)23に保持されデータアレイ(DA)24の読出しアドレスとして使用される。これよりデータアレイ24よりオペランドが読出され演算装置3に送られる。一方、R×タイプ命令の第1オペランドRRタイプ命令の第1、第2オペランドが格納されているレジスタ番号はレジスタ(IDR)18、デコード情報キュー(IDQ)19を経て演算装置3に送られ、レジスタ(EDR)31に保持される。演算装置3では演算実行に先立って命令制御装置1より送られたレジスタ番号をもとに汎用レジスタファイル(GR)32よりオペランドが読出される。読出されたレジスタ値のうち一方は第1オペランドとして演算レジスタ(DARO, FDARO)51、41に

演算装置3などで構成されている。

命令制御装置1は命令を記憶制御装置2より読出し、解釈し、必要ならばアドレス計算により論理アドレスを求めてオペランドを記憶制御装置2より読出し、操作コード、オペランド、操作情報等を演算装置3に転送する。演算装置3は命令制御装置1が設定した情報により演算を行ない、各種レジスタやステータスまたは記憶制御装置2を通じて主記憶の更新をおこなう。

命令制御装置1において、命令レジスタ(IR)11のあるビットフィールドにより汎用レジスタファイル(CGR)13からレジスタの内容を読出す。R×タイプの命令では、アドレス生成のために汎用レジスタファイル13によりインデックス値、ベース値が読出される。このインデックス値、ベース値は命令で直接指定されるディスプレースメント値とともにレジスタ(XR)15、(BR)16、(DR)14にそれぞれ設定される。これらのレジスタ14、15、16の値をもとにしてアドレス加算器(ADR)17よりオペ

設定される。もう一方のレジスタ値は記憶制御装置2より直接あるいはデータバッファ(DB)34を経由して送られたオペランドとセレクタ(SEL)33で選択され第2オペランドとして演算レジスタ(DAR1, FDAR1)52、42に設定される。演算ユニット35、36はそれぞれ2進基本演算、浮動小数点数演算をおこなう。演算ユニット(FAU)35は浮動小数点数演算をおこなうが、浮動小数点数演算は演算の前に浮動小数点数の桁合せを、後に正規化をおこなう必要があり、演算に複数のサイクル必要である。本実施例では被演算数の桁合わせで1サイクル、加減算で1サイクル、演算結果の正規化で1サイクル必要であり、浮動小数点数の演算には3サイクル必要である。一方、演算ユニット(BAU)36は2進基本演算をおこなうが、2進数演算は浮動小数点数演算に比べて簡単であり、本実施例では加減算1サイクルで終了する。なお、演算ユニット35は、レジスタ41、42、44、45、47、49、桁合せ回路43、浮動小数点数演算的

路(FALU)46、正規化回路48で構成され、演算ユニット36は、レジスタ51、52、54、55、56、論理演算回路(ALU)53で構成されている。選択回路37はPSWに付加された動作モードフラグに反応してレジスタ49、54、56の出力を選択して出力する。ある命令により、動作モードフラグに論理“1”を設定することで、演算実行のパイプラインに処理がおこなわれるが、演算ユニット35、36のパイプラインの長さをあわせるために2進基本演算の結果をもちまわるレジスタ(DCR、DDR)55、56が使用される。第2図(a)は本動作モードにおけるパイプラインのタイムチャートである。図において、D、A、P、C、L、E、N、Sはそれぞれデコードサイクル、アドレス生成サイクル、ページングサイクル、キャッシュリードサイクル、析合せサイクル、演算サイクル、正規化サイクル、ストアサイクルを示す。2進基本命令の演算実行サイクルも浮動小数点命令に合せてやはり3サイクルかけてパイプラインがスムーズに流れるように制御さ

れる。ある命令により、動作モードフラグに論理値“0”を設定することで演算実行のパイプライン処理はおこなわれないが、そのため演算ユニット35、36のパイプラインの長さをあわせる必要がないためレジスタ(DCR、DDR)55、56はバイパスされ、使用されない。第2図(b)は本動作モードにおけるパイプラインのタイムチャートである。第2図(b)に示すパイプラインは第2図(a)に示したものと異なり、2進基本命令のパイプラインではLサイクル、Nサイクルは存在せず演算処理サイクルはEサイクルの1サイクルのみである。しかし、浮動小数点命令では演算処理にL、E、Nサイクルの3サイクル必要であるため、後に2進基本命令が続く場合2サイクル空きが生じ、スループットが低下する。

第2図(c)、(d)は2つのパイプラインモードにおける分岐命令の処理(分岐予測失敗時)のタイムチャートである。分岐命令の分岐方向が決まるのは分岐命令の直前の命令の演算実行が終了したときである。分岐予測が成功したかどうかもち

の時点で決まり、予測失敗時はそこからデコードサイクルを始めることになる。第2図(c)に演算パイプラインモードでの分岐予測失敗のタイムチャートが示されている。同図で分岐命令の直前の2進基本命令の演算サイクルはNサイクルで終るが、このサイクルが終るまで分岐予測の成功/失敗がわからない。したがって、予測失敗の場合、分岐先命令の実行に5サイクルのおくれが生じてしまう。第2図(d)は動作モードフラグに論理値“0”が設定された非演算パイプラインモードでの分岐予測失敗のタイムチャートが示である。同図で分岐命令の直前の2進基本命令の演算処理サイクルはEサイクルで終るが、このサイクルが終ると分岐予測の成功/失敗が判る。したがって、分岐予測失敗の場合、分岐先命令の実行に3サイクルのおくれが生じるが、第2図(c)に示した場合よりおくれが2サイクル少ない。

このように、演算パイプラインモード動作では浮動小数点命令のように演算処理に複数サイクル必要な命令の処理でも余分なサイクルを必要とし

ないが、例えば分岐予測失敗によるおくれは大きい。一方、非演算パイプラインモード動作では浮動小数点命令のように演算処理に複数サイクル必要な命令の処理では演算処理サイクルにかかるだけパイプライン処理の乱れが生じるが、分岐予測失敗によるおくれは小さいという特徴がある。

本実施例では動作モードフラグが論理“1”のとき演算パイプラインモードになり論理値“0”のとき非演算パイプラインモードであったが、論理値の組合せは逆でも良い。

(発明の効果)

以上説明したように本発明は、演算実行のパイプライン処理をおこなわず、短いパイプライン動作をおこなう動作モードと、演算実行のパイプライン処理をおこない長いパイプライン動作をおこなう動作モードとの動作モード遷移をおこなうことにより、浮動小数点命令の頻度が低く、分岐命令の頻度の高いオペレーティングシステム等のシステムプログラム動作時は、非演算パイプライン動作をおこなうことで出現頻度の高い分岐命令の

4. 図面の簡単な説明

- 1 … 命令制御装置、
- 2 … 記憶制御装置、
- 3 … 演算装置、
- 3 5 … 浮動小数点数演算ユニット、
- 3 6 … 2進基本演算ユニット。

[illegible]

- 120 -

浮動小数点命令 : D A P C L E N S
 基本命令 : D A P C L E N S
 浮動小数点命令 : D A P C L E N S

(a)

基本命令 : D A P C E S
 浮動小数点命令 : D A P C L E N S
 基本命令 : D A P C E S
 2サイクル

(b)

基本命令 : D A P C L E N S
 分岐命令 : D A P C L E N
 分岐先命令 : D A P C L E N S
 5サイクル

(c)

基本命令 : D A P C E S
 分岐命令 : D A P C E
 分岐先命令 : D A P C E S
 3サイクル

(d)

第2図